**MLflow Alpha Release**

Warning

The current version of MLflow is an alpha. This means that APIs and storage formats are subject to change!

**Installing**

Install MLflow from PyPi via pip install mlflow

MLflow requires conda to be on the PATH for the projects feature.

**Documentation**

Official documentation for MLflow can be found at <https://mlflow.org/docs/latest/index.html>.

**Running a Sample App With the Tracking API**

The programs in example use the MLflow Tracking API. For instance, run:

python example/quickstart/test.py 需要增加一点东西 比如import Mlflow mlflow.set\_tracking\_uri(**".//log"**)

This program will use MLflow log API, which stores tracking data in ./mlruns, which can then be viewed with the Tracking UI.

**Launching the Tracking UI**

The MLflow Tracking UI will show runs logged in ./mlruns at [http://localhost:5000](http://localhost:5000/). Start it with:

mlflow ui

首先要进到有log产生的目录然后运行mlflow ui这样会产生mlruns目录 产生mlruns目录后再<http://localhost:5000>就能访问log信息

Mlruns目录下必须有很多记录信息时才能在网站上查找到

**Running a Project from a URI**

The mlflow run command lets you run a project packaged with a MLproject file from a local path or a Git URI:

mlflow run example/tutorial -P alpha=0.4

mlflow run git@github.com:databricks/mlflow-example.git -P alpha=0.4

mlflow run --no-conda https://github.com/cliicy/MLProjs.git -P alpha=5

mlflow run --no-conda https://github.com/databricks/mlflow-example.git -P alpha=8

\_fetch\_project:从本地fetch或者从github上fetch

\_run\_project

See example/tutorial for a sample project with an MLproject file.

**Saving and Serving Models**

To illustrate managing models, the mlflow.sklearn package can log Scikit-learn models as MLflow artifacts and then load them again for serving. There is an example training application in example/quickstart/test\_sklearn.py that you can run as follows:

$ python example/quickstart/test\_sklearn.py

Score: 0.666

Model saved in run <run-id>

$ mlflow sklearn serve -r <run-id> model

$ curl -d '[{"x": 1}, {"x": -1}]' -H 'Content-Type: application/json' -X POST localhost:5000/invocations